Genetic Algorithm Assisted HIDMS-PSO: A New Hybrid Algorithm for Global Optimisation

Fevzi Tugrul Varna
Department of Informatics, University of Sussex
Brighton, United Kingdom
f.varna@sussex.ac.uk

Phil Husbands
Department of Informatics, University of Sussex
Brighton, United Kingdom
philh@sussex.ac.uk

***Abstract*—In this paper, a new hybrid algorithm, GA-HIDMS-PSO, is introduced by hybridising the state-of-the-art particle swarm optimisation (PSO) variant, the heterogeneous improved dynamic multi-swarm PSO (HIDMS-PSO) with a genetic algorithm (GA). The new hybrid model exploits the heterogeneous features of HIDMS-PSO and the evolutionary characteristics of the GA. In the GA-HIDMS-PSO architecture, HIDMS-PSO acts as the primary search engine, and the GA is employed as the secondary method to assist and slow down the loss of diversity for selected proportions of homogeneous and heterogeneous subpopulations of the HIDMS-PSO algorithm. Both methods run consecutively. As the primary search method, HIDMS-PSO runs for longer periods compared with the GA. The HIDMS-PSO provides the initial solutions for the GA from both homogeneous and heterogeneous subpopulations and final solutions returned from the GA replace prior solutions in the HIDMS-PSO which resumes the search process with potentially more diverse particles to guide the swarm. The GA-HIDMS-PSO algorithm's performance was tested on the 30 and 50 dimensional CEC'05 and CEC'17 test suites. The results were compared with 24 algorithms, with 12 state-of-the-art PSO variants and 12 other metaheuristics. GA-HIDMS-PSO outperformed all 24 comparison algorithms on both test suites for both 30 and 50 dimensions.**

***Index Terms*—particle swarm optimisation, genetic algorithm, swarm intelligence, evolutionary algorithm, hybrid algorithm**

I. Introduction

Optimisation is a process of finding a feasible solution to a given problem under certain constraints. Although various practical methodologies are available for optimisation, the most predominant class of algorithms, metaheuristics, are frequently employed. The two most famous metaheuristics categories are evolutionary algorithms (EAs) and swarm intelligence algorithms (SIAs). The two most distinguished and widely applied algorithms from these classes are the GA and PSO, respectively. Both algorithms have many variants [1] [2] and applications [3] [4] in the literature. In the last decade, researchers turned towards a new and highly effective class of algorithms, namely the hybridisation of metaheuristics. The hybridisation of EAs with other types of algorithms is popular because of practicality and competence of EAs in dealing with uncertainty and noise. The problem of premature convergence is a core issue in the metaheuristics literature, and it mainly occurs due to lack of diversity. In a typical search process, initially, diversity is high, and depletion of diversity ensues as the population moves closer to the best-known optimum. Although in theory, high population diversity may help to guarantee finding the optimal solution, it may also result in slow convergence, meaning that an algorithm is in theory capable of finding the optimal solution but may never converge or meet the termination criteria in a reasonable timeframe. In contrast, in a search process with a low population diversity, fast convergence is usually observed with poor solution accuracy (convergence to local optima). The study [5] refers to the ideal balance between convergence and accuracy as the trade-off point. It is apparent that convergence is not guaranteed, even with sufficient diversity, but maintaining the balance of exploration and exploitation may boost an algorithm to perform at maximum capacity. To tackle this issue, hybridisation has become a widely accepted method to promote diversity during the search for the global optimum. HIDMS-PSO [6] is a state-of-the-art algorithm with a dynamic topological structure that possesses heterogeneous features and adopts several strategies to delay the loss of diversity in the population to tackle the problem mentioned above. In light of this, we aim to exploit the heterogeneous qualities of HIDMS-PSO, while extending its diversity-handling capabilities further, by hybridising it with a GA in a collaborative architecture, thus boosting particles' abilities to escape local optima. To maintain the aforementioned ideal trade-off point between convergence rate and accuracy, in our hybrid model, we combine the approach of sequential collaborative and partial manipulative integrative hybrid frameworks to efficiently exploit the heterogeneous features of HIDMS-PSO. In our model, the GA is employed for short periods (50 iterations) to assist HIDMS-PSO (which runs consecutively for 100 iterations) by evolving a proportion of both the homogeneous and heterogeneous subpopulations of the HIDMS-PSO. The sole purpose of this collaborative hybrid interaction is to prevent depletion of diversity within the population of HIDMS-PSO by periodically feeding subpopulations of HIDMS-PSO with the evolved solutions from GA. The evolved solutions returned from the GA are replaced with the positions (not $pbest$s) of the same particles from both subpopulations of the HIDMS-PSO. As a result, this causes fluctuations in the diversity of randomly selected proportions of both subpopulations. By only exchanging a proportion of both subpopulations between the two algorithms, we retain a significant fraction of agents unchanged in the HIDMS-

PSO algorithm. This strategy allows us to avoid the slow convergence issue while retaining diversity during the overall search, enabling convergence within a reasonable time to an accurate solution.

II. Related Studies

This section briefly introduces the required background on the canonical PSO algorithm, HIDMS-PSO and genetic algorithm.

A. Canonical PSO

PSO is a stochastic search algorithm based on the movement of a population of agents called particles. The swarm consists of N particles, and each particle has velocity $\vec{v}_i^{(t)}$, position $\vec{x}_i^{(t)}$ at time t and the personal best-known position $p\vec{best}$. Particles move in the n-dimensional search space by using three vectors, $\vec{v}_i^{(t-1)}$, $p\vec{best}$ and $g\vec{best}$, the position of the globally best particle in the swarm. The canonical PSO algorithm uses the following two equations to update particles' velocity and position:

$$\vec{v}_i^{(t+1)} = \omega\vec{v}_i^{(t)} + c_1\vec{r}_1(p\vec{best}_i - \vec{x}_i^{(t)}) + c_2\vec{r}_2(g\vec{best} - \vec{x}_i^{(t)}) \quad (1)$$

$$\vec{x}_i^{(t+1)} = \vec{x}_i^{(t)} + \vec{v}_i^{(t)} \quad (2)$$

Where ω is the inertia weight used to control the impact of the previous velocity, constants c_1 and c_2 control the attraction rate/level for cognitive ($p\vec{best}$) and social ($g\vec{best}$) attraction, $\vec{r}_1$ and $\vec{r}_2$ are random vectors $\in [0,1]^n$.

III. HIDMS-PSO

The HIDMS-PSO algorithm is a recent state-of-the-art PSO algorithm introduced by Varna and Husbands [6]. The algorithm introduced a new master-slave inspired dynamic topological structure with homogeneous and heterogeneous subpopulations and two distinct movement schemes, namely, inward-oriented and outward-oriented strategies. The small subswarm entities in the HIDMS-PSO algorithm are called units, and each unit constitutes one master particle and three slave particles with unique slave types. Particles maintain their assigned roles (e.g. master/slave) and the unit structure during the entire search process. The distinction in type between the slave particles allows heterogeneous particle behaviour, restricting rapid information exchange to prevent premature convergence and loss of population diversity. Fig. 1 shows the topological structure of a single unit.

Information flow and the way particles interact with one another has an immense impact on the population diversity and particles' guidance, hence the overall search process. The HIDMS-PSO algorithm employs a communication model to manage the exchange of information and the interaction between particles. The communication model restricts information flow and allows particles to exchange information through master-to-master and slave-to-slave communication (see Fig. 2). The main communication is governed by the following rules:

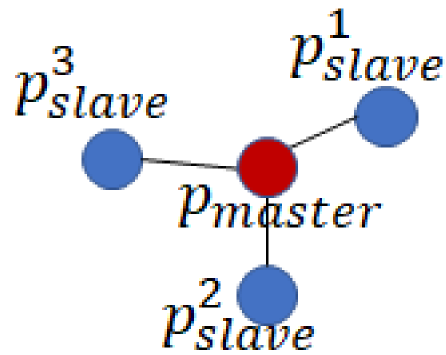

Fig. 1. The unit topological structure in HIDMS-PSO algorithm.

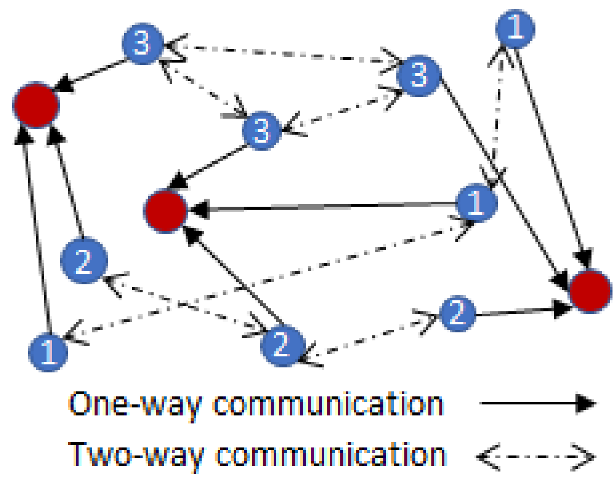

Fig. 2. The communication model, depicts the information exchange among three units.

1) Particles of a unit are not permitted to communicate with particles of another unit in the swarm. The exchange of positional information is only established via particles with slave roles.
2) Particles with master roles are only allowed to communicate with their corresponding slaves in the same unit.
3) Particles with slave roles are not allowed to communicate with slaves in the same unit. They are only permitted to exchange positional information with slave particles in other units of the same type.

1) Search Behaviour: In the HIDMS-PSO algorithm, the initial population is divided into two equal subpopulations, one homogeneous and one heterogeneous, and each subpopulation adopts a distinct movement strategy (Fig. 3). The homogeneous subpopulation uses the update equation of the canonical PSO algorithm, whereas the heterogeneous subpopulation is used to form N unit structures and adopts inward and outward-oriented strategies. The inward-oriented strategy aims to guide particles towards other members in the same unit. In contrast, the outward-oriented strategy guides particles using exemplars derived from different units.

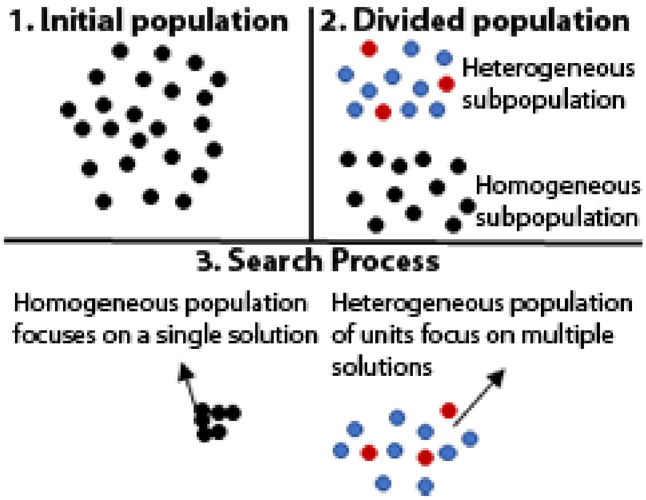

Fig. 3. Search phases of the HIDMS-PSO algorithm.

a) Inward-oriented strategy: The inward-oriented movement strategy exploits the positional information obtained from all unit members to provide guidance to particles. For the inward-oriented movement, particles with master roles update their velocities by randomly selecting one of the Equations. 3-5:

$$\vec{v}_m^{(t+1)} = \omega^{(t)}\vec{v}_m^{(t)} + c_1\vec{r}_1(\vec{pbest}_m - \vec{x}_m^{(t)}) + c_2\vec{r}_2(\vec{x}_s^{dis} - \vec{x}_m^{(t)}) \quad (3)$$

Where $\vec{v}_m^{(t)}$ is the velocity, $\vec{pbest}_m$ is the personal best position, $\vec{x}_m$ is the position of the master particle at time t and, $\vec{x}_s^{dis}$ is the positionally dissimilar slave particle in the unit N. The position of the master particle has a significant impact on slave particles. Hence, units aim to maintain their diversity by guiding the master particle towards $\vec{x}_s^{dis}$.

$$\vec{v}_m^{(t+1)} = \omega^{(t)}\vec{v}_m^{(t)} + c_1\vec{r}_1(\vec{pbest}_m - \vec{x}_m^{(t)}) + c_2\vec{r}_2(\vec{x}_s^{best} - \vec{x}_m^{(t)}) \quad (4)$$

Where $\vec{x}_s^{best}$ is the N^{th} unit's fittest slave particle's position. By moving the master particle in the direction of the fittest slave, the master particle carry out a local exploration.

$$\vec{v}_m^{(t+1)} = \omega^{(t)}\vec{v}_m^{(t)} + c_1\vec{r}_1(\vec{pbest}_m - \vec{x}_m^{(t)}) + c_2\vec{r}_2(\vec{x}_s^{avg} - \vec{x}_m^{(t)}) \quad (5)$$

Where $\vec{x}_s^{avg}$ is the average position of all slaves within the master's current unit. The slave particles are directed towards the slave's personal best-known position and the master particle's position, using Eq. 6.

$$\vec{v}_s^{(t+1)} = \omega^{(t)}\vec{v}_i^{(t)} + c_1\vec{r}_1(\vec{pbest}_s - \vec{x}_s^{(t)}) + c_2\vec{r}_2(\vec{x}_m - \vec{x}_s^{(t)}) \quad (6)$$

Where $\vec{v}_s^{(t)}$ is the velocity, $\vec{pbest}_s$ is the best position found by the particle at time t, $\vec{x}_s$ is the slave particle's position and, $\vec{x}_m$ is the N^{th} unit's master particle's position.

b) Outward-oriented strategy: In contrast to the inward-oriented movement strategy, the outward-oriented strategy provides particles with guidance from different units in the swarm. For this movement strategy, particles with master roles randomly select one of the following equations (7-9) to update their velocities:

$$\vec{v}_m^{(t+1)} = \omega^{(t)}\vec{v}_m^{(t)} + c_1\vec{r}_1(\vec{pbest}_m - \vec{x}_m^{(t)}) + c_2\vec{r}_2(\vec{x}_{unit}^{avg} - \vec{x}_m^{(t)}) \quad (7)$$

Where $\vec{v}_m^{(t)}$ is the velocity, $\vec{pbest}_m$ is the personal best position, $\vec{x}_m$ is the position of the master particle at time t and, $\vec{x}_{unit}^{avg}$ is the mean position of particles that belong to the N^{th} unit.

$$\vec{v}_m^{(t+1)} = \omega^{(t)}\vec{v}_m^{(t)} + c_1\vec{r}_1(\vec{pbest}_m - \vec{x}_m^{(t)}) + c_2\vec{r}_2(\vec{x}_{unit}^{m} - \vec{x}_m^{(t)}) \quad (8)$$

Where $\vec{x}_{unit}^{m}$ is the master particle's position from a randomly selected unit.

$$\vec{v}_m^{(t+1)} = \omega^{(t)}\vec{v}_m^{(t)} + c_1\vec{r}_1(\vec{x}^{avg} - \vec{x}_m^{(t)}) + c_2\vec{r}_2(\vec{x}_{unit}^{m} - \vec{x}_m^{(t)}) \quad (9)$$

Where $\vec{x}^{avg}$ is the particle's own unit members' mean position and $\vec{x}_{unit}^{m}$ is the randomly select unit's master particle position. As with the guidance of the slave particles in the inward-oriented movement scheme, in the outward-oriented strategy the slave particles use the following velocity update equation to move in the direction of a selected slave (of the same type) particle from a randomly chosen unit.

$$\vec{v}_s^{(t+1)} = \omega^{(t)}\vec{v}_s^{(t)} + c_1\vec{r}_1(\vec{pbest}_s - \vec{x}_s^{(t)}) + c_2\vec{r}_2(\vec{x}_{unit}^{rnd} - \vec{x}_s^{(t)}) \quad (10)$$

Where $\vec{v}_s^{(t)}$ is the velocity, $\vec{pbest}_s$ is the personal best position, $\vec{x}_s$ is the position of the slave particle and, $\vec{x}_{unit}^{rnd}$ is the randomly chosen unit's slave particle of the same kind.

By integrating both homogeneous and heterogeneous subpopulations, the HIDMS-PSO algorithm maintains the balance of exploration and exploitation. Concurrently, the inward and outward-oriented movement schemes trigger single-time behavioural fluctuations that enhance each unit's population diversity and assist escape from local optima [6].

IV. Genetic Algorithm

The genetic algorithm [7] [8], introduced by John Holland, is inspired by biological evolution based on Darwin's theory of natural selection. In the literature, many GA variants [1] have been introduced and are successfully applied to a broad spectrum of problems [3]. Compared to traditional optimisation methods, the GA has several noticeable advantages, including parallelism and the ability to handle complex optimisation problems. Despite these assets, genetic algorithms have certain potential disadvantages that require careful assessment, and which could otherwise significantly impact on the efficiency and efficacy of the search process. These include the formulation of the problem/fitness function, setting an appropriate population size and tuning of other parameters, such as the selection criteria, mutation rate and crossover. Despite these challenges, genetic algorithms remain one of the most prevalently applied evolutionary algorithms to diverse problems. The main phases of genetic algorithms comprise of selection, crossover, mutation and elitism.

V. The Proposed Algorithm: GA-HIDMS-PSO

The main idea behind hybridisation is to compensate for the drawbacks of one or both algorithms used for hybridisation to improve the search process. In this particular case, PSO's main disadvantage is premature convergence with underlying causes triggered by the loss of diversity due to rapid information flow between particles. Many variants in the literature, including HIDMS-PSO, studied in this paper introduced mechanisms to deal with the aforementioned issue successfully to a certain extent. The study [9] describes three typical PSO-GA hybrid approaches prevalently used; these could be summarised as:

1) Approach 1: Both PSO and GA run in parallel. The global best solution in PSO is unchanged for a specific interval, and crossover operation is performed on $gbest$ with a GA chromosome.
2) Approach 2: Mutation operator of GA is employed to improve particles with stagnated $pbest$.
3) Approach 3: Initial population of PSO is generated by GA, remaining subsequent iterations are equally run by GA and PSO. The first half of the iterations are executed by GA, then PSO presumes the search using the final solutions obtained from GA as initial solutions.

In a more recent study [5], hybrid algorithms are grouped into two main categories as collaborative hybrid and integrative hybrid approaches. The former methodology refers to combining two or more algorithms running in either a parallel or sequential manner with several frameworks including multi-stage, sequential and parallel. In this approach, the contributing weight of each algorithm can be assumed to be equal (50/50). The latter hybrid method refers to integrating one of the algorithms into the main/master algorithm as a subordinate. This model offers two approaches, namely, full manipulation and partial manipulation. In this case, the contributing weight of the second algorithm is around 10 to 20%.

In our approach, HIDMS-PSO and GA are run consecutively and continuously for short periods until the total numbers of iterations are reached. The hybrid model employed in this study combines features of both the collaborative and the integrative hybrid frameworks. The collaborative interaction and consecutive executions of both algorithms are derived from the collaborative framework's sequential structure. On the other hand, GA's role in the collaborative relationship to evolve a proportion of both subpopulations is adopted from the integrative hybrid framework's partial-manipulation approach. A preliminary experiment to determine the optimum number of iterations to assign to each algorithm indicated that 100 iterations of HIDMS-PSO followed by 50 iterations of the GA gave the best result. The HIDMS-PSO algorithm is the primary

Algorithm 1: GA-HIDMS-PSO

population size n, dimensions d, $C = 0.15$, ω_{max}=0.99, ω_{min}=0.2;
randomly define each particle's velocity υ and position x;
$c_1 = 2.5 - (1 : T_{max} * 2/T_{max})$;
$c_2 = 0.5 - (1 : T_{max} * 2/T_{max})$;
$\omega_1(t) = \frac{\omega_{max} + (\omega_{min} - \omega_{max})}{1 + exp\left(-5\left(\frac{2t}{T_{max}} - 1\right)\right)}$;
RG_{min} = T_{max}*0.01;
RG_{max} = T_{max}*0.1;
RG = RG_{max}; $phase_1 = 100$;
for t=1:T_{max} **do**
  if $t < T_{max}$*0.9)==0 **then**
    if *mod(t,$phase_1$)==0* **then**
      GA=true;
    end
  end
  if *mod(t,T_{max}*RG)==0* **then**
    vertically shuffle slave particles
  end
  if *mod(t,T_{max}*0.05)==0* **then**
    if $t < T_{max} * 0.9$ **then**
      β = round(d*U(0.1,1));
    else
      β = round(d*0.1);
    end
    for *j=1:n* **do**
      select β number of random dimensions to mutate for each particle
    end
  end
  if *GA==false* **then**
    for *i=1:n* **do**
      if $f(x_i) >= \overline{f(x)}$ **then**
        ω = $\omega_1^{(t)}$ + C; if $\omega > 0.99$, ω = 0.99 end;
      else
        ω = $\omega_1^{(t)}$ - C; if $\omega < 0.20$, ω = 0.20 end;
      end
      if *randi([0 1])==0 (inward-strategy)* **then**
        if *i^{th} particle is a master* **then**
          behaviour = randi([1 3]);
          if *behaviour == 1* **then**
            update υ_i and x_i using Eqs. 4 and 2
          else if *behaviour == 2* **then**
            update υ_i and x_i using Eqs. 5 and 2
          else if *behaviour == 3* **then**
            update υ_i and x_i using Eqs. 6 and 2
          end
        else
          update υ_i,x_i using Eqs. 7 and 2
        end
      else
        if *i^{th} particle is a master* **then**
          behaviour = randi([1 3]);
          if *behaviour == 1* **then**
            update υ_i,x_i using Eqs. 8 and 2
          else if *behaviour == 2* **then**
            update υ_i,x_i using Eqs. 9 and 2
          else if *behaviour == 3* **then**
            update υ_i,x_i using Eqs. 10 and 2
          end
        else
          update υ_i,x_i using the Eqs. 11 and 2
        end
      end
      perform partial non-uniform mutation on the x_i
      Evaluate the fitness of x_i
      Update the p_{best} and g_{best}
      i^{th} particle communicates according to the rules stated in section 3
      $RG = round(RG_{max} - (RG_{max} - RG_{min}) * \frac{t}{T_{max}})$
    end
  else
    pop_1 = Random N/2 particles from homogeneous pop
    pop_2 = Random N/2 particles from heterogeneous pop
    $GA_{initialPop}$ = [pop_1 pop_2];
    $GA_{finalSols}$ = GeneticAlgorithm($GA_{initialPop}$);
    replace $GA_{finalsols}$ with the positions of same particles
    update g_{best}
    GA=false;
  end
end

search method in our hybridisation model, and the GA is used to reverse or slow down the depletion of diversity by evolving a sub-population of the current HIDMS-PSO swarm. As part of our preliminary experiment, we employed various strategies to determine which particles should be passed onto and used as initial solutions by the genetic algorithm. We experimented with using the whole population, continuously feeding the same set of particles from the same subpopulation (whether the homogeneous or heterogeneous population), selecting the least fit subpopulation on average and selecting only master or slave particles as initial solutions. Although few of these selection methods were found to produce satisfactory results, it was discovered that selecting half of both the homogeneous and heterogeneous sub-populations provide the optimal performance for our hybrid model. The GA-HIDMS-PSO algorithm initiates the search process with 100 iterations of HIDMS-PSO, then half of both the homogeneous and heterogeneous populations are randomly selected, and their positions (not their $pbest$) are provided to the GA as initial solutions. The indices of those particles are preserved to update them in the next step. With the initial solutions provided, the GA runs for 50 iterations and returns the evolved final solutions to replace the same particles' positions in the HIDMS-PSO's population. The cycle repeats. It is worth noting that the hybridisation approach 2 mentioned at the start of this section suggests the application of a mutation operation on $pbest$ to improve stagnated particles. However, in our hybridisation approach the use of $pbest$ instead of the particle's current position resulted in deterioration and much better performance was observed when GA-returned solutions replaced the current positions instead of $pbest$. Although the investigation of this issue is not within the scope of this study, it is anticipated that the deterioration is related to the similarity of the solutions returned from the GA which are then used to update $pbest$ values causing a sudden loss of diversity in proportions of both subpopulations. On the other hand, updating particles' current positions triggers fluctuations in the evolved particles' positions, potentially contributing to particles' escape from local optima, and as a result improve their $pbest$. The GA-HIDMS-PSO algorithm uses the same parametric settings as the standard HIDMS-PSO; for a detailed description of the parameters, refer to the original study [6]. Besides the standard PSO parameters c_1, c_2 and ω, the HIDMS-PSO employ an additional parameter RG to reshape the unit structures at specific intervals (see pseudocode).

VI. Experimental Results and Discussions

In this section, we discuss the experimental setup and the corresponding results.

1) Experimental Setup: We have carried out three experiments to assess the performance of the GA-HIDMS-PSO algorithm on the CEC'05 [10] and CEC'17 [11] benchmark test suites. The CEC'17 benchmark test suite includes 30, and the CEC'05 suite consists of 25 test functions with diverse characteristics. For the first and second experiments, we replicated the experiments conducted in [6] and for the third experiment, study [12] was replicated to produce comparable results. In the first experiment, the performance of the GA-HIDMS-PSO algorithm is tested using the CEC'17 test suite. The results of the GA-HIDMS-PSO algorithm is compared with 11 baseline methods: two canonical PSO algorithms with distinct parameters ($\omega = 0.9 \rightarrow 0.4, c_1, c_2 = 2$ and $\omega = 0.4, c_1, c_2 = 2$), and evolutionary algorithms (the bat algorithm (BA) [13] ($A = 0.25, r = 0.5, f_{min}, f_{max} = 0.2$), grey wolf optimiser (GWO) [14] ($a_0 = 2$), butterfly optimisation algorithm (BOA) [15], whale optimisation algorithm (WOA) [16], moth flame optimisation (MFO) [17], artificial bee colony (ABC) [18], flower pollination algorithm (FPA) [19] ($p = 0.8$), cuckoo search algorithm (CS) [20] ($p = 0.25$) and invasive weed optimisation (IWO) [21]). In the second experiment, GA-HIDMS-PSO's performance was tested using the CEC'05 test suite and results were compared against 6 state-of-the-art PSO variants: HIDMS-PSO [6] ($\omega = 0.99 \rightarrow 0.29, c_1 = 2.5 \rightarrow 0.5, c_2 = 0.5 \rightarrow 2.5, RG_{min} = T_{max} * 0.01, RG_{max} = T_{max} * 0.1$), HCLDMS-PSO [22] ($\omega = 0.99 \rightarrow 0.29, c_1 = 2.5 \rightarrow 0.5, c_2 = 0.5 \rightarrow 2.5, Pm = 0.1$), FDR-PSO [23], HCLPSO [24], HPSO-TVAC [25], MNHPSO-JTAC [26] and for the third experiment, results were also compared against 6 state-of-the-art PSO variants: CLPSO [27] ($\omega = 0.9 \rightarrow 0.2, c_1, c_2 = 1.49445, V_{max} = 0.2 * Range$), DMSPSO [28], χPSO [29] (ring with neighborhood radius $n_r = 2, \phi = 4.1, \chi = 0 : 72984, c_1, c_2 = 2.05$), BBPSO [30], ($\omega = 0.729, c_1, c_2 = 1.49445, V_{max} = 0.5 * Range$), FIPS [31] and UPSO [32]. In the first experiment, the population size was set to 100 for all metaheuristics, and 40 for the two PSO variants and GA-HIDMS-PSO. In the second and third experiment, for all PSO variants, the population was set as 40 [6] [12]. For the first two experiments, each problem in the test suite was run 30 times, and in the third experiment, 100 times for 300,000 function evaluations at 30 dimensions and 500,000 function evaluations at 50 dimensions. For detailed parameter values for comparison algorithms used in the experiments, refer to [6] [12] and the original studies. Table I-VI display the mean errors obtained for the first experiment conducted on the CEC'17 test suite for 30 and 50-dimensional problems. The average and final ranks of the mean performances obtained for all three experiments are displayed in Table VII-IX. The Wilcoxon signed-rank test is conducted to statistically evaluate the achieved results for the GA-HIDMS-PSO algorithm. For the first experiment, at 30 dimensions, the result is not significant for only HIDMS-PSO, and at 50 dimensions, the result is significant for all comparative methods. For the second experiment conducted on the CEC'05 test suite, the result is significant for both 30 and 50 dimensions. Finally, for the third experiment, the result is not significant for only two algorithms, namely CLPSO and BBPSO at $p < 0.05$ for problem size of 30 and 50 dimensions. Due to the length restrictions of this paper, experimental results are partially included. External supplementary material is provided for complete results of experiments that can be accessed from users.sussex.ac.uk/fv47/GA-HIDMS-PSO.pdf.

A. Results

The CEC'17 test suite's experimental results at 30 dimensions reveal that the proposed algorithm (GA-HIDMS-PSO) outperformed all comparison methods for 13 of the 29 problems. HIDMS-PSO and CS outperformed all comparison algorithms in a total of 5 problems, while ABC achieved the best mean performance in 6 problems. The same experiment conducted at 50 dimensions showed that GA-HIDMS-PSO achieved the best performance for 18 problems. The CS algorithm achieved the best performance for 6 and HIDMS-PSO in 4 of the 29 problems. The second experiment is conducted using the CEC'05 test suite, and the results at 30 dimensions revealed that GA-HIDMS-PSO outperformed comparison methods for 11 problems. HCLDMS-PSO attained the best mean performance for 6, HCLPSO for 3, HPSO-TVAC for 2 and HIDMS-PSO for a single problem. The same experiment conducted at 50 dimensions reveals that GA-HIDMS-PSO attained the best result for 17 of 25 problems. HCLPSO outperformed all methods for 4 problems, while HIDMS-PSO and HCLDMS-PSO attained the best performance in two cases. The third experiment is also conducted using the CEC'05 test suite. The results at 30 dimensions reveal that GA-HIDMS-PSO outperformed comparison methods for 9, and CLPSO achieved the best mean results for 10 problems. BBPSO outperformed comparison algorithms for 7 problems while χPSO and DMSPSO attained the best performance for single a problem. The same experiment conducted at 50 dimensions reveals that GA-HIDMS-PSO attained the best mean results for 8 problems, followed by BBPSO for 7 problems and CLPSO, which obtained the best performance for 6 problems. DMSPSO and UPSO both outperformed comparison methods for 3 problems.

The impact of the new hybrid model on population diversity and convergence was assessed by running HIDMS-PSO and GA-HIDMS-PSO twenty times consecutively on the CEC'17 problems F1, F5, F10, F15, F20 and F25 at 30 dimensions. Fig. 4 and Fig. 5 shows the recorded average diversity and convergence rate for both algorithms. In Fig 4, it is observed that in each case, GA-HIDMS-PSO maintained significantly better population diversity for the entire search period until the last exploitation phase of the search process. The periodic fluctuations observed in the diversity rate of GA-HIDMS-PSO is an indicative of the GA-returned solutions causing sudden improvements in the diversity. The convergence rates shown in Fig. 5 indicates that GA-HIDMS-PSO is capable of converging at a faster rate to a better solution in comparison to HIDMS-PSO.

VII. Conclusions

This study proposed a new hybrid algorithm GA-HIDMS-PSO for global optimisation by hybridising genetic algorithm with state-of-the-art HIDMS-PSO. The hybrid model is designed to allow the GA to assist HIDMS-PSO to further improve the diversity maintaining capabilities and convergence rate of the HIDMS-PSO algorithm. In our approach, both algorithms' roles can be summarised as, HIDMS0-PSO being the primary search method that controls the main population and the search. GA is the secondary algorithm employed to evolve the appointed particles selected from both homogeneous and heterogeneous subpopulations of HIDMS-PSO to improve particles' diversity in both sub-populations continuously. The proposed algorithm was tested on CEC'05 and CEC'17 test suites against 12 metaheuristics and 12 state-of-the-art PSO variants at 30 and 50 dimensions. The comparison revealed the superiority of the GA-HIDMS-PSO on both test suites. In addition, the comparison of diversity and convergence rate between hybrid version and HIDMS-PSO revealed significant improvements in diversity, convergence and the quality of solution found. The present work may be further extended by improving the performance of GA-HIDMS-PSO; alternatively, the algorithm can be applied to practical real-world and noisy problems.

References

[1] B. M. Varghese and R. J. S. Raj, "A survey on variants of genetic algorithm for scheduling workflow of tasks," 2016 Second International Conference on Science Technology Engineering and Management (ICONSTEM), Chennai, 2016, pp. 489-492, doi: 10.1109/ICONSTEM.2016.7560870.

[2] M. Imran, R. Hashim, and N. E. A. Khalid, "An overview of particle swarm optimization variants," Procedia Eng., vol. 53, pp. 491-496, 2013.

[3] C. K. H. Lee, "A review of applications of genetic algorithms in operations management," Eng. Appl. Artif. Intell., 2018, doi: 10.1016/j.engappai.2018.08.011.

[4] Y. del Valle, G. K. Venayagamoorthy, S. Mohagheghi, J. C. Hernandez, and R. G. Harley, "Particle swarm optimization: Basic concepts, variants and applications in power systems," IEEE Transactions on Evolutionary Computation. 2008, doi: 10.1109/TEVC.2007.896686.

[5] Ting T.O., Yang XS., Cheng S., Huang K. (2015) Hybrid Metaheuristic Algorithms: Past, Present, and Future. In: Yang XS. (eds) Recent Advances in Swarm Intelligence and Evolutionary Computation. Studies in Computational Intelligence, vol 585. Springer, Cham.

[6] F. T. Varna and P. Husbands, "HIDMS-PSO: A New Heterogeneous Improved Dynamic Multi-Swarm PSO Algorithm," 2020 IEEE Symposium Series on Computational Intelligence (SSCI), Canberra, Australia, 2020, pp. 473-480, doi: 10.1109/SSCI47803.2020.9308313.

[7] De Jong K, Analysis of the behavior of a class of genetic adaptive systems, Ph.D. thesis, University of Michigan, Ann Arbor, MI, USA; 1975.

[8] Holland J. Adaptation in natural and artificial systems. Ann Arbor, MI, USA: University of Michigan Press; 1975.

[9] Premalatha, K.; Natarajan, A.M. Hybrid PSO and GA for global maximization. Int. J. Open Probl. Comput. Math. 2009, 2, 597-608.

[10] P.N. Suganthan, N. Hansen, J.J. Liang, Y.P. Chen, K. Deb, A. Auger, S. Tiwari, Problem definitions and evaluation criteria for the CEC 2005 special session on real-parameter optimization, Technical Report: Nanyang Technological University AND KanGAL Report (2005005), 2005.

[11] N.H. Awad, M.Z. Ali, P.N. Suganthan, J.J. Liang, B.Y. Qu, Evaluation Criteria for the CEC 2017 Special Session and Competition on Single Objective Real-Parameter Numerical Optimization, 2016, Tech. Rep.

[12] M. G. Epitropakis, V. P. Plagianakos, and M. N. Vrahatis, "Evolving cognitive and social experience in Particle Swarm Optimization through Differential Evolution: A hybrid approach," Inf. Sci. (Ny)., 2012, doi: 10.1016/j.ins.2012.05.017.

[13] X.-S. Yang, A new metaheuristic bat-inspired algorithm, Nature Inspired Cooperative Strategies for Optimization (NICSO 2010), Springer, 2010.

[14] S. Mirjalili, S.M. Mirjalili, A. Lewis, Grey Wolf Optimizer, Adv. Eng. Software 69 (2014) 46-61.

[15] S. Arora, S. Singh, Butterfly optimisation algorithm: a novel approach for global optimisation, Soft Comput. 23 (3) (2018) 715-734.

[16] S. Mirjalili, A. Lewis, The Whale Optimization Algorithm, Adv. Eng. Software 95 (2016) 51-67.

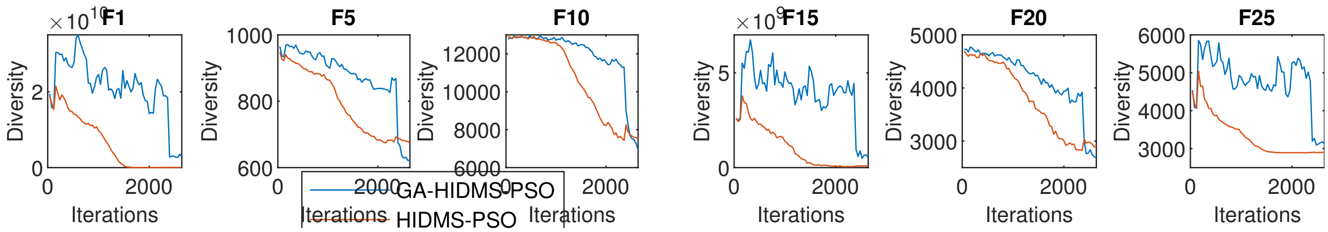

Fig. 4. Rate of diversity comparison for HIDMS-PSO and GA-HIDMS-PSO.

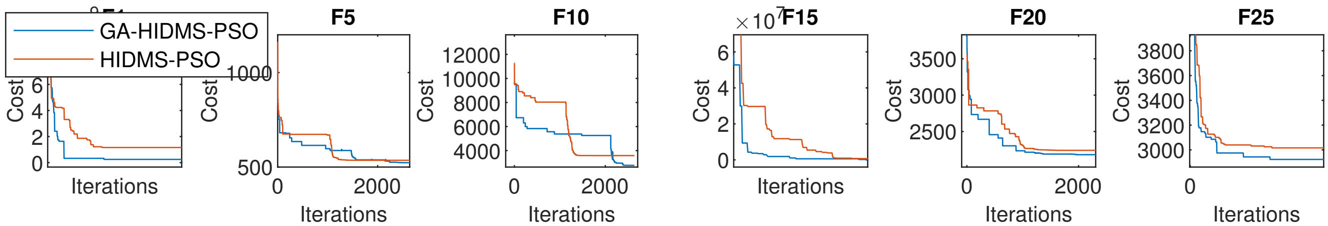

Fig. 5. Convergence rate comparison for HIDMS-PSO and GA-HIDMS-PSO.

TABLE I
THE RESULTS OBTAINED FOR THE FIRST EXPERIMENT CONDUCTED USING THE CEC'17 TEST SUITE FOR PROBLEM SIZE OF 30 DIMENSIONS.

	F1	F3	F4	F5	F6	F7	F8	F9	F10
BA	7.3E+10	2.2E+05	2.1E+04	5.1E+02	1.1E+02	1.5E+03	4.3E+02	2.1E+04	8.8E+03
GWO	1.1E+09	2.9E+04	1.5E+02	8.7E+01	4.0E+00	1.6E+02	7.7E+01	5.4E+02	2.8E+03
BOA	3.0E+10	6.7E+04	2.5E+03	3.3E+02	6.4E+01	5.1E+02	2.9E+02	6.9E+03	7.7E+03
WOA	2.1E+06	1.6E+05	1.5E+02	2.7E+02	6.6E+01	5.1E+02	1.9E+02	7.7E+03	4.8E+03
MFO	8.1E+09	7.7E+04	5.1E+02	1.8E+02	2.5E+01	3.5E+02	1.7E+02	5.1E+03	4.1E+03
ABC	1.3E+02	1.2E+05	3.4E+01	8.8E+01	0.0E+00	1.0E+02	8.9E+01	8.2E+02	2.3E+03
FPA	1.1E+11	1.8E+06	3.6E+04	6.2E+02	1.3E+02	2.5E+03	5.6E+02	3.1E+04	9.1E+03
CS	1.9E+04	4.5E+04	7.5E+01	1.4E+02	5.0E+01	1.6E+02	1.3E+02	4.6E+03	3.7E+03
IWO	3.0E+03	6.4E+03	8.8E+01	4.1E+02	7.2E+01	2.0E+03	3.5E+02	7.6E+03	4.7E+03
PSO_1	1.3E+11	3.9E+08	4.4E+04	6.8E+02	1.4E+02	2.7E+03	6.1E+02	3.8E+04	9.6E+03
PSO_2	1.3E+11	3.9E+08	4.4E+04	6.8E+02	1.4E+02	2.7E+03	6.1E+02	3.8E+04	9.6E+03
HIDMS-PSO	4.7E+03	0.0E+00	6.1E+01	5.2E+01	0.0E+00	8.7E+01	4.8E+01	2.6E+00	2.7E+03
GA-HIDMSPSO*	2.8E+03	4.4E-09	6.7E+01	3.8E+01	6.5E-03	7.8E+01	3.8E+01	1.7E+00	2.5E+03

TABLE II
THE RESULTS OBTAINED FOR THE FIRST EXPERIMENT CONDUCTED USING THE CEC'17 TEST SUITE FOR PROBLEM SIZE OF 50 DIMENSIONS.

	F1	F3	F4	F5	F6	F7	F8	F9	F10
BA	1.7E+11	8.2E+07	6.3E+04	9.5E+02	1.3E+02	3.3E+03	9.7E+02	7.5E+04	1.6E+04
GWO	4.6E+09	7.0E+04	4.3E+02	1.7E+02	1.1E+01	3.0E+02	2.0E+02	3.7E+03	5.6E+03
BOA	4.3E+10	2.2E+05	9.9E+03	6.2E+02	7.9E+01	1.1E+03	6.5E+02	2.8E+04	1.4E+04
WOA	7.1E+06	7.8E+04	2.8E+02	4.2E+02	7.6E+01	9.9E+02	4.1E+02	1.9E+04	9.1E+03
MFO	3.2E+10	1.7E+05	2.6E+03	4.2E+02	4.5E+01	9.0E+02	3.8E+02	1.5E+04	7.9E+03
ABC	9.2E+08	6.6E+05	1.2E+03	5.0E+02	3.0E+01	5.7E+02	5.0E+02	3.0E+04	1.5E+04
FPA	2.3E+11	1.9E+08	9.0E+04	1.1E+03	1.4E+02	4.7E+03	1.1E+03	9.2E+04	1.6E+04
CS	1.4E+05	1.6E+05	7.7E+01	2.9E+02	6.2E+01	3.4E+02	2.8E+02	1.6E+04	7.0E+03
IWO	6.9E+03	2.6E+04	1.2E+02	7.4E+02	7.8E+01	3.5E+03	7.2E+02	2.0E+04	7.7E+03
PSO_1	1.3E+09	9.6E+03	2.5E+02	2.3E+02	2.0E+01	2.8E+02	2.3E+02	5.8E+03	6.5E+03
PSO_2	1.2E+10	5.8E+04	9.3E+02	2.0E+02	1.2E+01	2.7E+02	2.0E+02	3.6E+03	6.1E+03
HIDMS-PSO	5.4E+03	0.0E+00	7.0E+01	1.1E+02	1.2E-01	1.7E+02	1.1E+02	5.6E+01	5.6E+03
GA-HIDMSPSO*	4.9E+03	4.2E-03	7.0E+01	8.7E+01	7.6E-02	1.6E+02	7.9E+01	3.3E+01	4.8E+03

[17] S. Mirjalili, Moth-flame optimisation algorithm: A novel nature-inspired heuristic paradigm, Knowl.-Based Syst. 89 (2015) 228-249. M.A. Awadallah.

[18] D. Karaboga, B. Basturk, A powerful and efficient algorithm for numerical function optimization: artificial bee colony (ABC) algorithm, J. Global Optim. 39 (3) (2007) 459-471.

[19] Yang, X. (2016). Nature-inspired optimisation algorithms.: Elsevier, pp.261-263.

[20] Yang, X. (2016). Nature-inspired optimisation algorithms.: Elsevier, pp.256-258.

[21] Mehrabian, A. and Lucas, C., 2006. A novel numerical optimisation algorithm inspired from weed colonisation. Ecological Informatics, 1(4), pp.355-366.

[22] Wang, S., Liu, G., Gao, M., Cao, S., Guo, A. and Wang, J., 2020. Heterogeneous comprehensive learning and dynamic multi-swarm particle swarm optimiser with two mutation operators. Information Sciences, 540, pp.175-201.

[23] T. Peram, K. Veeramachaneni, C.K. Mohan, Fitness-distance-ratio based particle swarm optimisation, in: Proceedings of the 2003 IEEE Swarm Intelligence Symposium, IEEE, 2003, pp. 174-181.

[24] N. Lynn, P.N. Suganthan, Heterogeneous comprehensive learning particle swarm optimisation with enhanced exploration and exploitation, Swarm Evol. Comput. 24 (2015) 11-24.

[25] A. Ratnaweera, S.K. Halgamuge, H.C. Watson, Self-organising hierarchical particle swarm optimiser with time-varying acceleration coefficients, IEEE Trans. Evol. Comput. 8 (3) (2004) 240-255.

[26] Ghasemi, M., Akbari, E., Zand, M., Hadipour, M., Ghavidel, S. and Li, L., 2019. An Efficient Modified HPSO-TVAC-Based Dynamic Economic Dispatch of Generating Units. Electric Power Components and Systems, 47(19-20), pp.1826-1840.

[27] Liang, J., Qin, A., Suganthan, P. and Baskar, S., 2006. Comprehensive learning particle swarm optimiser for global optimisation of multimodal functions. IEEE Transactions on Evolutionary Computation, 10(3), pp.281-295.

[28] J.-J. Liang, P.N. Suganthan, Dynamic multi-swarm particle swarm optimiser, in: Proceedings of the IEEE Swarm Intelligence Symposium,

TABLE III

THE RESULTS OBTAINED FOR THE SECOND EXPERIMENT CONDUCTED USING THE CEC'05 TEST SUITE FOR PROBLEM SIZE OF 30 DIMENSIONS.

	F1	F2	F3	F4	F5	F6	F7	F8	F9	F10
HIDMS-PSO	1.4E-12	1.1E-03	1.1E+06	1.7E+03	3.0E+03	7.0E+01	4.7E+03	2.1E+01	5.0E+01	6.5E+01
HPSO-TVAC	5.5E-14	4.8E-02	1.7E+06	3.0E+03	5.5E+03	1.1E+02	4.7E+03	2.1E+01	3.6E+01	1.0E+02
FDR	5.0E+02	1.4E+03	1.6E+07	2.8E+03	3.6E+03	2.4E+06	4.7E+03	2.1E+01	2.7E+02	2.0E+02
HCLDMS-PSO	3.3E-12	3.5E+01	2.9E+06	2.2E+03	2.8E+03	6.3E+01	4.7E+03	2.1E+01	3.7E+01	3.5E+01
HCLPSO	1.3E+01	2.2E+01	3.7E+06	2.1E+03	2.4E+03	2.9E+05	4.7E+03	2.1E+01	4.0E+00	6.7E+01
MNHPSO-JTVAC	5.9E-14	9.3E-03	9.8E+05	3.6E+03	5.4E+03	9.9E+01	4.7E+03	2.1E+01	2.5E+01	1.0E+02
GA-HIDMS-PSO*	2.1E-13	1.1E-09	5.4E+05	1.5E+02	1.7E+03	4.9E+01	4.7E+03	2.1E+01	1.3E+01	4.3E+01

TABLE IV

THE RESULTS OBTAINED FOR THE SECOND EXPERIMENT CONDUCTED USING THE CEC'05 TEST SUITE FOR PROBLEM SIZE OF 50 DIMENSIONS.

	F1	F2	F3	F4	F5	F6	F7	F8	F9	F10
HIDMS-PSO	2.5E-09	2.8E+01	3.8E+06	2.5E+04	6.8E+03	1.2E+02	6.2E+03	2.1E+01	1.2E+02	1.3E+02
HPSO-TVAC	1.0E-13	1.9E+02	4.4E+06	3.1E+04	1.6E+04	1.7E+02	6.2E+03	2.1E+01	1.1E+02	1.9E+02
FDR	1.3E+03	1.1E+04	7.2E+07	2.6E+04	8.2E+03	9.9E+06	6.2E+03	2.1E+01	5.6E+02	4.3E+02
HCLDMS-PSO	6.9E-07	2.8E+03	1.1E+07	2.2E+04	7.5E+03	2.4E+02	6.2E+03	2.1E+01	1.1E+02	9.5E+01
HCLPSO	8.0E+00	2.0E+03	1.4E+07	2.5E+04	6.3E+03	1.8E+05	6.2E+03	2.1E+01	1.8E+01	1.2E+02
MNHPSO-JTVAC	1.2E-13	9.6E+01	2.9E+06	2.7E+04	1.4E+04	1.3E+02	6.2E+03	2.1E+01	8.3E+01	1.6E+02
GA-HIDMS-PSO*	0.0E+00	0.0E+00	1.0E+06	4.8E+03	4.2E+03	7.3E+01	6.2E+03	2.1E+01	5.5E+01	8.1E+01

TABLE V

THE RESULTS OBTAINED FOR THE THIRD EXPERIMENT CONDUCTED USING THE CEC'05 TEST SUITE FOR PROBLEM SIZE OF 30 DIMENSIONS.

	F1	F2	F3	F4	F5	F6	F7	F8	F9	F10
χPSO	9.7E+00	1.6E+01	1.0E+07	1.8E+03	8.1E+03	1.2E+03	6.8E+03	2.1E+01	6.5E+01	8.7E+01
BBPSO	0.0E+00	9.3E-03	1.3E+06	2.3E+03	5.3E+03	2.8E+01	4.7E+03	2.1E+01	5.6E+01	7.6E+01
DMSPSO	3.1E+02	7.8E+02	5.6E+06	8.6E+02	4.3E+03	2.7E+07	4.3E+03	2.1E+01	4.8E+01	8.0E+01
FIPS	5.3E+02	1.5E+04	1.9E+07	2.1E+04	1.2E+04	2.5E+07	7.5E+03	2.1E+01	5.4E+01	1.5E+02
UPSO	1.3E+03	7.6E+03	5.3E+07	1.9E+04	1.3E+04	1.2E+07	7.5E+03	2.1E+01	7.8E+01	1.6E+02
CLPSO	0.0E+00	3.8E+02	1.2E+07	5.4E+03	4.0E+03	1.8E+01	4.7E+03	2.1E+01	0.0E+00	8.0E+01
GA-HIDMS-PSO*	1.6E-03	1.2E+01	3.9E+06	8.4E+02	2.4E+03	1.8E+02	4.4E+03	2.1E+01	4.0E+00	4.4E+01

TABLE VI

THE RESULTS OBTAINED FOR THE THIRD EXPERIMENT CONDUCTED USING THE CEC'05 TEST SUITE FOR PROBLEM SIZE OF 50 DIMENSIONS.

	F1	F2	F3	F4	F5	F6	F7	F8	F9	F10
χPSO	9.7E+00	7.8E+02	2.0E+07	2.8E+04	1.1E+04	6.4E+06	6.2E+03	2.1E+01	1.8E+02	1.8E+02
BBPSO	0.0E+00	2.9E+02	3.7E+06	3.0E+04	1.3E+04	5.8E+01	6.2E+03	2.1E+01	1.3E+02	1.8E+02
DMSPSO	3.9E+02	9.7E+02	1.3E+07	1.3E+04	5.5E+03	1.8E+07	6.1E+03	2.1E+01	9.9E+01	1.7E+02
FIPS	1.7E+03	2.6E+04	5.9E+07	3.4E+04	1.6E+04	8.0E+07	1.0E+04	2.1E+01	1.5E+02	3.9E+02
UPSO	7.1E+02	4.2E+03	5.3E+07	1.4E+04	1.2E+04	2.7E+06	7.4E+03	2.1E+01	6.5E+01	1.4E+02
CLPSO	0.0E+00	1.0E+04	4.9E+07	3.4E+04	9.7E+03	8.7E+01	6.2E+03	2.1E+01	0.0E+00	2.2E+02
GA-HIDMS-PSO*	2.2E-01	1.1E+03	1.6E+07	1.0E+04	5.9E+03	2.6E+03	6.2E+03	2.1E+01	1.9E+01	9.9E+01

TABLE VII

RANKS OF MEAN PERFORMANCE FOR THE FIRST EXPERIMENT.

Algorithm	Avg(30D)	Final(30D)	Avg(50D)	Final(50D)
GA-HIDMS-PSO*	1.93	1	1.52	1
HIDMS-PSO	2.17	2	2.14	2
ABC	2.93	3	8.45	10
CS	3.90	4	4.17	3
GWO	5.24	5	5.00	4
MFO	6.48	6	8.10	9
IWO	6.86	7	7.10	7
WOA	7.21	8	7.83	8
BOA	8.03	9	9.69	11
BA	10.03	10	12.03	12
FPA	10.90	11	12.93	13
PSO$_1$	11.97	12	5.48	5
PSO$_2$	11.97	12	6.55	6

TABLE VIII

RANKS OF MEAN PERFORMANCE FOR THE SECOND EXPERIMENT.

Algorithm	Avg(30D)	Final(30D)	Avg(50D)	Final(50D)
GA-HIDMS-PSO*	2.04	1	1.40	1
HCLDMS-PSO	3.00	2	3.68	4
HIDMS-PSO	3.16	3	3.29	2
HCLPSO	3.68	4	3.56	3
HPSO-TVAC	4.48	5	4.76	6
MNHPSO-JTVAC	4.52	6	4.32	5
FPR	6.24	7	6.32	7

TABLE IX

RANKS OF MEAN PERFORMANCE FOR THE THIRD EXPERIMENT.

Algorithm	Avg(30D)	Final(30D)	Avg(50D)	Final(50D)
GA-HIDMS-PSO*	2.08	1	2.1	1
CLPSO	2.32	2	3.32	2
BBPSO	2.8	3	3.44	4
χPSO	3.8	4	4.2	6
DMSPSO	4.16	5	3.32	2
FIPS	5.84	6	6.4	7
UPSO	6.04	7	4.12	5

IEEE, 2005, pp. 124-129.

[29] Clerc, M. and Kennedy, J., 2002. The particle swarm - explosion, stability, and convergence in a multidimensional complex space. IEEE Transactions on Evolutionary Computation, 6(1), pp.58-73.

[30] Kennedy, J., n.d. Bare bones particle swarms. Proceedings of the 2003 IEEE Swarm Intelligence Symposium. SIS'03 (Cat. No.03EX706),.

[31] Mendes, R., Kennedy, J. and Neves, J., 2004. The Fully Informed Particle Swarm: Simpler, Maybe Better. IEEE Transactions on Evolutionary Computation, 8(3), pp.204-210.

[32] K.E. Parsopoulos, M.N. Vrahatis, UPSO: a unified particle swarm optimization scheme, in: Lecture Series on Computer and Computational Sciences: Proc. of Int. Conf. of Computational Methods in Sciences and Engineering, 2004, pp. 868-873.